SOFTWARE ARTS TECHNICAL NOTES, BY THE CREATORS OF VISICALC

A Publication of
Software Arts, Inc.

SATN

THE JOURNAL FOR VISICALC USERS

Vol. I, No. 6
July/August 1982

IN THIS ISSUE:

About This Issue 1

An Introduction to Boolean Functions 1
- Boolean Values
- Boolean Functions: Using @AND and @OR
- Nesting Booleans within @IF
- Calculating Commissions and Bonuses with Booleans

The VisiCalc Planner 4

Related Products Checklist 9
- Templates
- Graphics
- Communications
- Word Processors
- Other Software
- Hardware
- Training

From Our Readers 10

Index 12

. . . Plus VisiCalc Pointers

ABOUT THIS ISSUE

Recent issues of SATN have concentrated on applications of the VisiCalc® program. In this issue of SATN we will deal less with applications than with tools for using the program more efficiently.

In *An Introduction to Boolean Functions* we document some of the uses of @AND and @OR, two of the VisiCalc program's built in logical functions. The article deals mainly with the basics of these functions, and how they can be used with @IF.

While we were visiting with a SATN reader one afternoon last winter, he mentioned that it would be handy to have a "map" of the VisiCalc sheet to keep track of where different things were located in a file, and for planning a sheet. It struck us as an interesting idea and a potentially useful tool, so we've created one for you.

You'll also find a *Related Products Checklist* in this issue, the first installment of what we hope will be an ongoing column appearing in alternate issues.

We are also introducing a *From Our Readers* column, though it is not a "letters to the editor" column in the classic sense. That is, it is not a forum for the expression of opinions, good or bad. So you won't see any praise there for SATN or the VisiCalc program, just as you won't see any pans. What you will find are specific answers to interesting questions that one or more readers have asked about the use of the program. We'll also print brief notes from readers if they contain ideas that we think other readers will find useful. In short we shall try to keep it practical and brief.

Finally, this issue of SATN brings Volume I to a close, and so we have included an index of the volume.

AN INTRODUCTION TO BOOLEAN FUNCTIONS

We all use Boolean mathematics every day and rarely think of it. Many "to buy or not to buy" decisions, for example, involve Boolean considerations even before they involve arithmetic ones. That is, among the first questions asked about such decisions is "Can we afford it?" By asking that question we ask if the money available is greater than or less than the money needed; we compare quantities rather than adding or subtracting them. And since we have made a comparison, the question is answered *true* or *false*, and not numerically.

So, when the solution to a problem entails comparing values, it may very well call for the use of the logical, or Boolean functions (after George Boole, a 19th century English mathematician) of the VisiCalc® program. In this article we will examine @AND and @OR, two of the program's Boolean functions. Elsewhere in this issue there is a letter from a reader in which we show how to use another, @ISERROR.

Boolean Values

When Boolean functions are entered on a VisiCalc sheet, comparison operators are used analogously to arithmetic operators (+, −, *, /). In the VisiCalc program there are three comparison operators: greater than (>); less than (<), and equal to (=). Those operators can be combined to allow still more comparisons to be made: greater than or equal to is > = ; less than or equal to is < = ; not equal to is < >.

The comparisons result in Boolean *values* of TRUE or FALSE. In some programs the Boolean values TRUE and FALSE may be represented by 1 and 0, respectively. Strictly speaking, that is incorrect since TRUE and FALSE are conditions, not quantities, and cannot be used directly in arithmetic operations. That is, the statement "4 divided by TRUE" does not make sense and should not be taken to mean the same thing as "4 divided by 1."

One of the reasons for using TRUE and FALSE in the VisiCalc program was to catch errors before they could be propagated throughout a model. Thus, if you were to enter 3>2 on a VisiCalc sheet you would get TRUE. If you then multiplied TRUE by some number, the program would return ERROR. Similarly, the entry 5>4>3 also returns ERROR. If 1 or 0 had been used for TRUE and FALSE, such statements as TRUE∗4 or 5>4>3, though nonsensical, might be evaluated.

Note that although the words TRUE and FALSE appear on the screen they are *values, not labels.* Should you wish, you may enter @TRUE or @FALSE on a sheet and they will be treated as values, meaning among other things that they can be returned by @LOOKUP or @CHOOSE or counted by @COUNT. Boolean values cannot be used in @SUM, @AVERAGE, @MIN, @MAX, or @INT, all of which require numerical values.

In some applications the values compared may be single numbers, as in expressions like 1<2 or B1>C1, either of which can be evaluated directly and will return TRUE or FALSE. In other applications, entire expressions or functions may be substituted for single values. For example, A1∗.6 – A2 could be compared to @SUM(C2...F2). In that case, the expression (A1∗.6 – A2) = @SUM(C2...F2) might be used.

When only one comparison is made, it can be done directly, as in the expressions shown in the preceding paragraph. There are times, though, when it's necessary to make a series of comparisons. When that is the case, the Boolean functions of the VisiCalc program should be brought to bear on the problem.

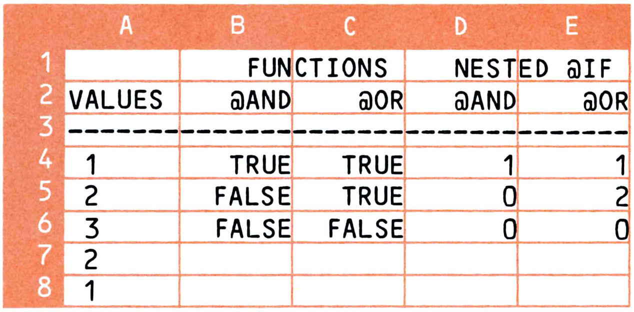
Boolean Functions: Using @AND and @OR

The @AND function compares one or more Boolean values and if any one is false returns FALSE. Only if all are true will the @AND function return TRUE.

The @OR function also compares one or more Boolean values, but if any one is true returns TRUE. Only if all are false will the @OR function return FALSE.

To examine the difference between @AND and @OR, set up a VisiCalc sheet with the labels shown in Figure 1. Set the column width to seven characters with the command:

```
/GC7
```



```
     A      B      C      D      E
1           FUNCTIONS     NESTED @IF
2 VALUES   @AND    @OR   @AND    @OR
3 -----------------------------------
4 1        TRUE   TRUE      1      1
5 2       FALSE   TRUE      0      2
6 3       FALSE  FALSE      0      0
7 2
8 1
```

FIGURE 1. *Try changing the values in column A to read 1 through 5 in ascending order and then in descending order. Another interesting variation is 2, 4, 3, 5, 1.*

In A4...A8 enter 1, 2, 3, 2, 1. For simplicity's sake we will ask only whether any of the values in column A are less than (<) the other values. Enter:

```
>B4:@AND(A4<A5,A5<A6)
```

Since 1 (A4) is less than 2 (A5), and 2 (A5) is less than 3 (A6), the @AND function returns TRUE.

Now enter:

```
>C4:@OR(A4<A5,A5<A6)
```

Although the same comparison is made with the @AND and @OR statements, and both return TRUE, there is a difference that is not apparent from these results alone. To see the difference, we can replicate those entries into rows 5 and 6. Enter:

```
>B4:/R...C4:B5...B6:RRRRRRRR
```

By making all value references relative, we compare the following conditions: 2<3, 3<2, and 2<1. Only the first of those comparisons is true, and it is in row 5 that we first see the difference. Because one of the two comparisons made is true and the other false, the @AND statement returns FALSE at B5, but the @OR statement returns TRUE at C5.

SATN is published bi-monthly by Software Arts, Inc., P.O. Box 494, Cambridge, Massachusetts 02139. Address all editorial correspondence to the Editorial Department, SATN, P.O. Box 494, Cambridge, MA 02139. Requests for special permission or bulk orders should be addressed to the publisher.

Domestic subscriptions: 6 issues, $30.00; outside U.S.A., $50.00. Single copy price, $6.00. Send subscriptions, change of address, and fulfillment questions to SATN Subscriptions, P.O. Box 815, Quincy, MA 02169.

Copyright © 1982, Software Arts, Inc. All rights reserved. This periodical may not be copied or reproduced in any fashion either in whole or in part without the prior written consent of Software Arts, Inc.

Software Arts, Inc. makes no warranty, express or implied, with respect to the quality, accuracy or freedom from error of the information contained in SATN, the journal for VisiCalc users, and specifically disclaims all liability for any damages resulting from the use of such information.

Unless specifically stated otherwise in any article in this issue of SATN, Software Arts, Inc. has not examined any products and/or publications identified in SATN to determine their accuracy and/or ability to perform as advertised. Mention of a product and/or publication in SATN does not constitute an endorsement by Software Arts, Inc. of any such product or publication.

Apple is a registered trademark of Apple Computer, Inc. Atari is a trademark of Atari Inc. Commodore is a trademark of Commodore International. DIF is a trademark of Software Arts, Inc. HP-83/85 and HP-125 are trademarks of Hewlett-Packard Company. IBM is a registered trademark of International Business Machines Corporation. TRS-80 is a trademark of Radio Shack Division of Tandy Corp.

SATN is a trademark of Software Arts, Inc.
VisiCalc is a registered trademark of VisiCorp/Personal Software Inc.

Nesting Booleans within @IF

Nesting @AND or @OR within an @IF can increase the power and versatility of the @IF function since it expands the number of things that can be compared. In the real world, choices often have to be made after comparing more than one thing to one other. If nothing can take place until every condition is met, that is, if a series of threshold values must all be reached before any action can be taken, an @AND nested within an @IF statement would be called for. On the other hand, if crossing one of several thresholds is sufficient to make a decision, then you would use an @OR nested within the @IF expression. Although an @IF expression can contain both an @AND and an @OR function, in the interest of keeping things simple, it's probably best to avoid doing so within a single expression.

Columns D and E in Figure 1 contain @IF statements within which we have nested @AND and @OR statements. The basic form of the nested @IF expressions we shall examine is:

@IF (@AND(returns true), then return the value in column A, else return 0)

@IF(@OR(returns true), then return the value in column A, else return 0)

The specific expressions on the sheet shown are:

```
>D4:@IF(@AND(A4<A5,A5<A6),A4,0)
>E4:@IF(@OR(A4<A5,A5<A6),A4,0)
```

Replicate those expressions into rows 5 and 6 with:

```
>D4:/R...E4:D5...D6:RRRRRRRRRR
```

Now if you return to column A and substitute different values (1, 2, 3, 4, 5, for example, or 5, 4, 3, 2, 1) you can see the differing effects of the comparisons made by @AND and @OR.

For those who prefer the concrete to the abstract, let's consider a more immediate, though hypothetical, application in which @AND and @OR can be used within an @IF statement. One place where such statements can be used effectively is in calculating commissions and bonuses.

Calculating Commissions and Bonuses with Booleans

Starting with a clear sheet, enter:

```
/GFI
/GC7
```

to set the display to integers and the column width to seven characters.

Here are the ground rules for an imaginary company's sales force. First, to earn an annual commission, each salesperson must generate at least $500,000 volume. Second, the company has two product lines, one big ticket, and one small ticket. To prevent the sales force from concentrating on the one to the exclusion of the other, the volume of the low priced line must exceed 60% of the sales of the high priced line. Third, careful records are kept of each account's payment history, and a delinquency factor assigned to the account. If any salesperson's weighted delinquency factor is more than 25%, no commission is paid. (We won't consider how such a factor might be derived, but its purpose is clearly to discourage sales to firms that don't pay their bills.)

Besides commissions, the company pays annual bonuses. To qualify for a bonus, a salesperson must open an average of one new account each month, or surpass his or her preceding year's volume by at least 20%. Since we've already made things complicated, we'll keep the commissions and bonuses simple. Commissions equal 3% of sales, bonuses 1%.

Rows 1...4 consist of labels, as does column A. Enter the labels shown in Figure 2, and enter the sales volumes for each territory as shown in columns B and C, rows 5...9. In column D, TOTAL SALES is the sum of sales in the HIGH LINE and LOW LINE. Enter and replicate:

```
>D5:+B5+C5®/R:D6...D9:RR
```

The figures shown for DELINQ FACTOR are entered as values, formatted with /FG. Columns F and G, 1980 VOLUME and '81 NEW ACCTS are also entered values, which you should copy from the figure.

The first set of calculations on columns A...G, other than the sum taken in column D, begins at H5 in the column headed COMM for commissions. Since

KEY TO CONVENTIONS USED IN SATN

The VisiCalc® program is available on personal computers manufactured by Apple, Atari, Commodore, Hewlett-Packard, IBM, and Radio Shack. To reduce any confusion arising from the differences among the keyboards of those computer systems, SATN uses a standard symbol and name to represent certain commonly used keys.

The symbol ® and the word RETURN are used as synonyms for the keys known as ENTER, ENDLINE, and RETURN. The symbol ^ is used to represent an exponentiation operator.

To make VisiCalc documentation more readable, the program was designed so that the colon (:) has the same effect as RETURN in some commands, as when ending the > command or separating various elements of the replicate command.

In examples of what to type, type exactly what you see and press RETURN at the end of every line. Users of the HP-83/85 may find it necessary to add parentheses to some of the expressions included in the text. For example, 3+4*5 should be (3+4)*5 on an HP-83/85.

Throughout SATN, one typeface represents both VisiCalc input and output (what you should type and what you will see on the screen). The typeface looks like this:

```
This line is typeset in OCR-B.
So is this: +A1+B1*3.25
```

To allow you to type the formulas directly on a VisiCalc sheet, examples in SATN specify coordinates, like A1, rather than the names of the values they represent. When actually typing a formula, you can use the arrows to point rather than using the > command.

there are three tests for earning a commission, each of which must be met in order to qualify, we use @AND. The first test, whether sales in the territory are at least $500,000, is performed with a "greater than or equal to" operator (> =), The second test, whether sales of the low line exceed 60% of sales of the high line, is performed with a "greater than" operator (>). The third test, whether delinquencies are less than 25%, uses a "less than" operator (<). If all three tests are successfully "passed" the @IF condition is TRUE and the salesperson is eligible for a commission of 3% of sales. If any test is failed, FALSE is returned, and no commission is paid. Enter:

```
>H5:@IF(@AND(D5>=500000,C5/B5>.6,
  E5<.25),D5*.03,0)
```

Replicate that expression into the rows below. Enter:

```
>H5:/R:H6...H9:RRRRR
```

Given the values in columns B...E, we can see that whoever is responsible for territory #3 has failed to earn a commission. The obvious reason is because sales in that territory did not reach $500,000. If sales in that territory had gone over $500,000, though, and had remained proportionally the same as shown in the figure, there would still have been no commission paid, because the LOW LINE equals only .599 of the HIGH LINE. (Check that by entering +C7/B7 somewhere on the sheet.)

Payment of a bonus only requires passing one of two tests, both of which are tested with "greater than or equal to" operators (> =). Enter:

```
>I5:@IF(@OR(G5>=12,D5>=(1.2*F5)),
  D5*.01,0)
```

Replicate that expression with the entry:

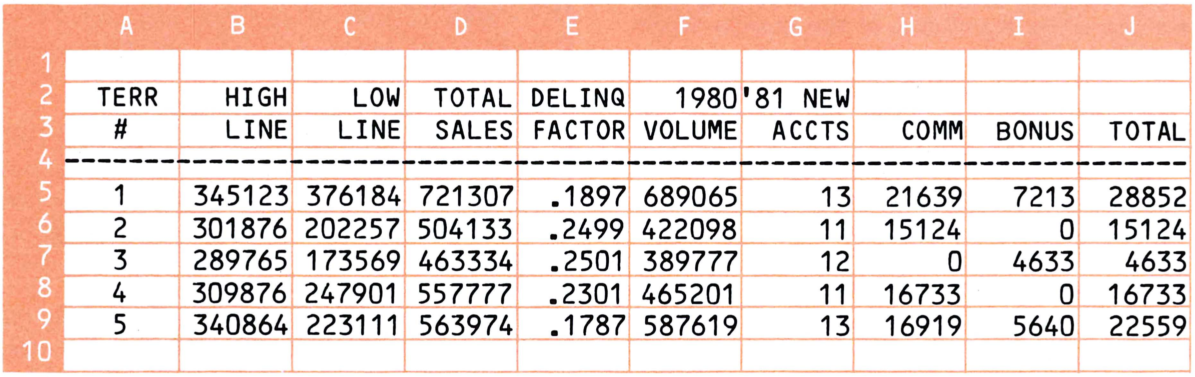
```
>I5:/R:I6...I9:RRRR
```

Since the salespersons in territories #2 and #4 did not earn bonuses, they must have failed to produce sales increases of at least 20%, since they also did not open at least 12 new accounts. In fact, none of the territories showed a 20% increase, and had the bonus calculation used an @AND statement, no bonuses would have been paid.

Finally, the TOTAL incentive payment for each territory is calculated beginning at J5. Enter:

```
>J5:+H5+I5®/R:J6...J9:RR
```

Calculating commissions and bonuses is only one of many ways to use the logical functions of the VisiCalc program. Among the other practical applications that come to mind are using @AND to distribute the dollar values of invoices into categories to determine the proportion of a company's bills that cost more to process than they can produce in profit. A similar application of @AND can be employed to distribute test grades into ranges to produce a marking curve for a class of students. In such applications the @AND function would test whether A is greater than B, but less than C (A>B, A<C). In future issues, we'll look more specifically at some of the other uses of Boolean functions, including how they can be used in calculating depreciation under the new tax law.



	A	B	C	D	E	F	G	H	I	J
1										
2	TERR	HIGH	LOW	TOTAL	DELINQ	1980	'81 NEW			
3	#	LINE	LINE	SALES	FACTOR	VOLUME	ACCTS	COMM	BONUS	TOTAL
4	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------
5	1	345123	376184	721307	.1897	689065	13	21639	7213	28852
6	2	301876	202257	504133	.2499	422098	11	15124	0	15124
7	3	289765	173569	463334	.2501	389777	12	0	4633	4633
8	4	309876	247901	557777	.2301	465201	11	16733	0	16733
9	5	340864	223111	563974	.1787	587619	13	16919	5640	22559
10										

FIGURE 2. *An imaginary company's commission and bonus calculation.*

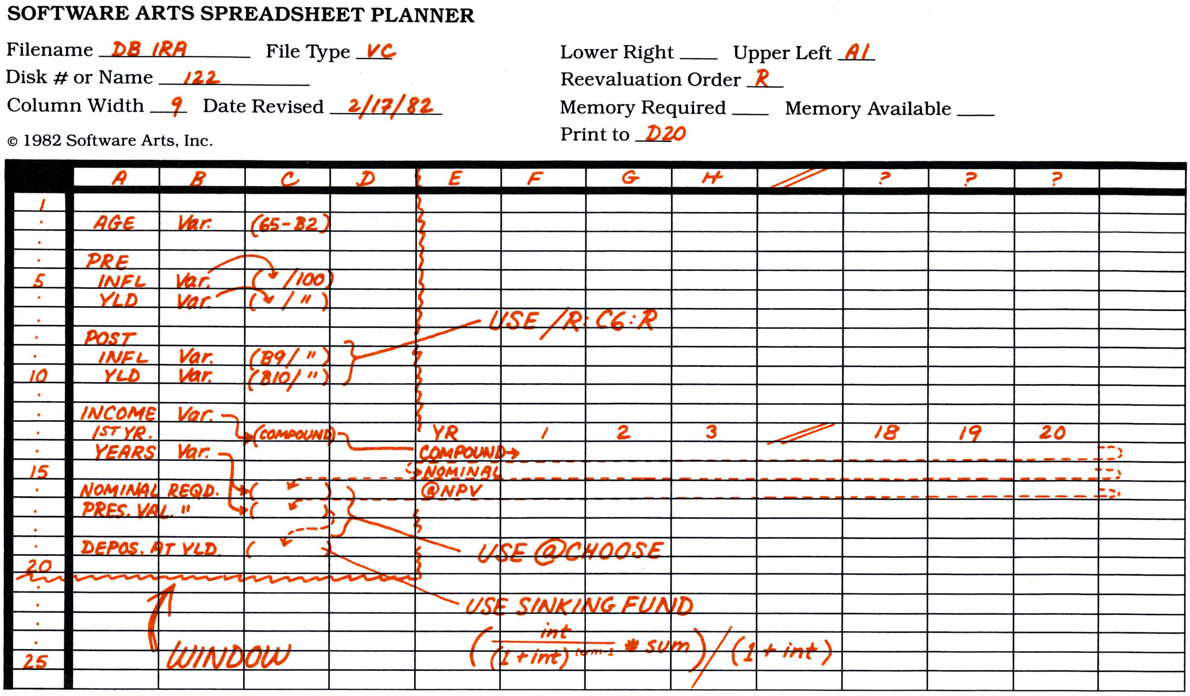
THE VISICALC PLANNER

It's not uncommon to hear from readers who have created models that take 30 seconds or more to recalculate completely. If the sheet has to be recalculated two or three times in order to get a correct answer, those thirty-second waits somehow seem to add up to more than a minute or a minute and a half.

What often happens in such cases is that rows or columns have been inserted in models without careful planning to avoid forward references. Or, as the sheet is developed, some elements of the model are entered in an order that would calculate efficiently if the sheet were set to reevaluate by rows, but the sheet is in fact reevaluating by columns.

The VisiCalc® Planner, reproduced as an 11" by 17" blank sheet on pages six and seven, should provide you with a tool that will minimize such problems by allowing you to sketch out a rough "map" of a particular model before you actually enter it in your VisiCalc program. We often use such maps as we develop models for SATN. They are invaluable planning aids, especially if the model is particularly complex.

Figures 3 and 4 illustrate how the VisiCalc Planner was used with one of the sheets in the IRA article in SATN Vol I, No. 4. The figures show the VisiCalc Planner superimposed on the screen output for the sheet.



SOFTWARE ARTS SPREADSHEET PLANNER

Filename DB IRA File Type VC
Disk # or Name 122
Column Width 9 Date Revised 2/17/82
Lower Right ___ Upper Left A1
Reevaluation Order R
Memory Required ___ Memory Available ___
Print to D20

© 1982 Software Arts, Inc.

	A	B	C	D	E	F	G	H	⁄⁄	?	?	?
1												
.	AGE	Var.	(65-B2)									
.												
.	PRE											
5	INFL	Var.	(↓ /100)									
.	YLD	Var.	(↓ / ")									
.												
.	POST											
.	INFL	Var.	(B9/ ")									
10	YLD	Var.	(B10/ ")									
.												
.	INCOME	Var.										
.	1ST YR.		(COMPOUND)		YR	1	2	3	⁄⁄	18	19	20
.	YEARS	Var.			COMPOUND→							
15					NOMINAL							
.	NOMINAL	REQD.	(		@NPV							
.	PRES. VAL.	"	(									
.												
.	DEPOS. AT	YLD.	(									
20												
.												
.												
.												
.												
25												

USE /R: C6:R

USE @CHOOSE

USE SINKING FUND

$\left(\frac{int}{(1+int)^{term-1}} * sum\right)/(1+int)$

WINDOW

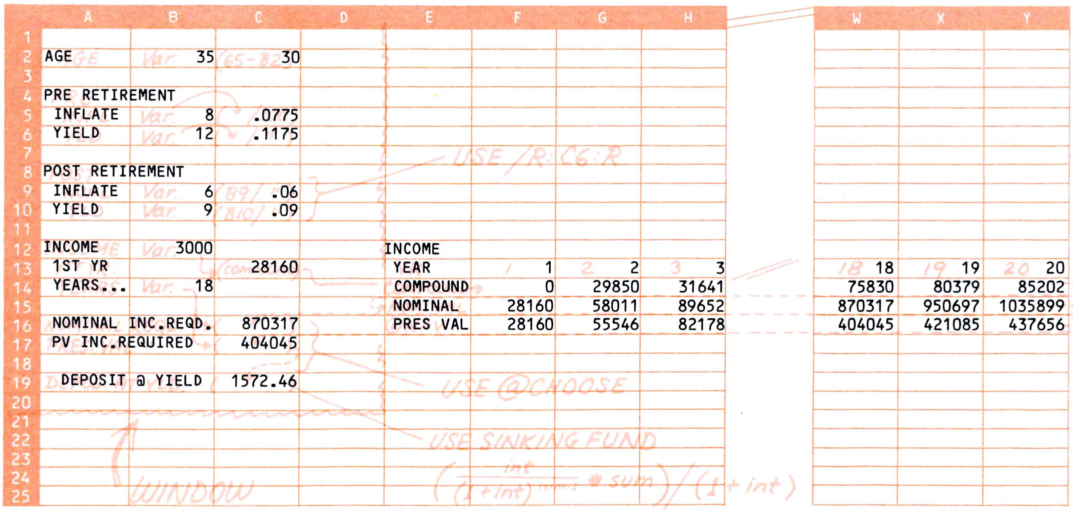
FIGURES 3 and 4. *These two figures should be compared carefully. Figure 3 shows the plan for a sheet. Figure 4 shows the sheet superimposed on the plan.*

One of our objectives when developing the model used for the "defined benefit IRA" shown in Vol. I, No. 4, was to create a sheet that could be printed as a one page report that included any assumptions about inflation and investment yield, but not showing year-by-year details. We also decided that the sheet should fit within a single window so that if a printed report was not necessary, all the conclusions reached could be displayed without having to scroll any part of the sheet out of sight. Since the model was developed on an Apple II, the window is defined with D20 at the lower right. We defined the window on the VisiCalc planning sheet shown in Figure 3, although it was not strictly necessary to do so. If the sheet were constructed so that the key information on the screen is to be located in some place other than the actual upper left corner, then sketching in the window area as we have done here becomes more important.

Notice that all the variables are placed above and to the left of the part of the sheet where the year-by-year calculations of income are made. In fact, the rows on which those calculations may be made most efficiently are determined by the location of variables in Column B.

In Figure 3 the arrows, whether solid or dashed, show the "flow" of the key calculations on the sheet. The only reason for drawing them differently is to make them slightly easier to read against the background of the grid.

The advantage of a tool like the VisiCalc Planner is not only that it maps a model, but also that it can serve as the basis for the model's documentation. Note that the remarks introduced by the word "use" make some preliminary decisions about functions and expressions to be used. For example, the general formula for a sinking fund, the calculation required at C19, has been written directly on the Planner. If it is ever necessary for you to discuss the model with someone who is unfamiliar with it, it can be helpful to lay out its general structure first. That is especially so if the listener is also unfamiliar with the VisiCalc program.



	A	B	C	D	E	F	G	H	W	X	Y
1											
2	AGE	35	30								
3											
4	PRE RETIREMENT										
5	INFLATE	8	.0775								
6	YIELD	12	.1175								
7											
8	POST RETIREMENT										
9	INFLATE	6	.06								
10	YIELD	9	.09								
11											
12	INCOME	3000			INCOME						
13	1ST YR		28160		YEAR	1	2	3	18	19	20
14	YEARS...	18			COMPOUND	0	29850	31641	75830	80379	85202
15					NOMINAL	28160	58011	89652	870317	950697	1035899
16	NOMINAL INC.REQD.		870317		PRES VAL	28160	55546	82178	404045	421085	437656
17	PV INC.REQUIRED		404045								
18											
19	DEPOSIT @ YIELD		1572.46								
20											
21											
22											
23											
24											
25											

SOFTWARE ARTS SPREADSHEET PLANNER

Filename ______ File Type ___
Disk # or Name ______
Column Width ___ Date Revised ______

© 1982 Software Arts, Inc.

Lower Right ___ Upper Left ___
Reevaluation Order ___
Memory Required ___ Memory Available ___
Print to ___

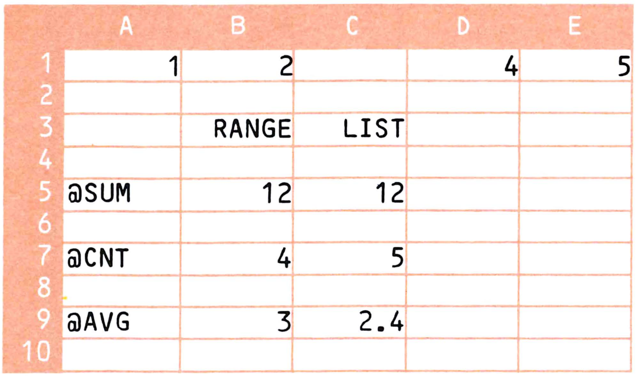
© 1982 Software Arts, Inc.

© 1982 Software Arts, Inc.

VISICALC POINTER

@COUNT and @AVERAGE are used to evaluate either ranges or lists of entries. When you use @COUNT or @AVERAGE to evaluate a range, a blank cell is treated differently from the way it is treated in a list.

When a range is specified in either of those functions, then the contents of each cell in the range are evaluated. Since a blank cell contains nothing, it is treated as though it does not exist.



	A	B	C	D	E
1	1	2		4	5
2					
3		RANGE	LIST		
4					
5	@SUM	12	12		
6					
7	@CNT	4	5		
8					
9	@AVG	3	2.4		
10					

FIGURE 5. *The expressions on this sheet are:*

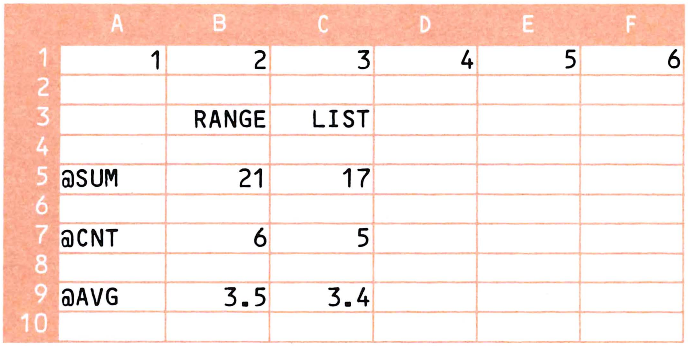
```
>B5:@SUM(A1...E1)
>B7:@COUNT(A1...E1)
>B9:@AVERAGE(A1...E1)
>C5:@SUM(A1,B1,C1,D1,E1)
>C7:@COUNT(A1,B1,C1,D1,E1)
>C9:@AVERAGE(A1,B1,C1,D1,E1)
```

Enter those expressions on a clear sheet after entering the series 1 . . . 5 in A1 . . . E1. If you then blank C1 your sheet should look exactly like the one in the figure. Notice that the sums at B5 and C5 are always equal. The counts at B7 and C7 will vary, however, depending on whether there are any blank cells between A1 and E1, as will the averages. That is because @AVERAGE is @SUM/@COUNT. If you blank all of the cells from A1 through E1, the @AVG calculation at B9 will return ERROR because of division by zero.

If a list is specified, then the expressions in the cells are evaluated. Since a blank is considered to be an expression with the value of zero, the cell is treated as though it does exist.

Because of the differing treatment of lists and ranges you must plan carefully how you will specify the values that you want to have averaged with @AVERAGE. In Figure 5 we show the way that the different treatment of a blank can affect the same operation using the same values.

One of the considerations to bear in mind when you are planning your sheets is whether or not you will be inserting rows or columns. If you think that you will be inserting them, but do not wish to have the inserted material evaluated in the @COUNT or @AVERAGE calculations, then specify lists. If you wish to have the inserted rows or columns included in the evaluations, then specify ranges (see Figure 6).



	A	B	C	D	E	F
1	1	2	3	4	5	6
2						
3		RANGE	LIST			
4						
5	@SUM	21	17			
6						
7	@CNT	6	5			
8						
9	@AVG	3.5	3.4			
10						

FIGURE 6. *If you have created the sheet described in the caption to Figure 5, insert a column with >D1:/IC and change the values at D1 . . . F1 to conform to this figure. Those expressions specifying ranges include the newly inserted column D, while those that specify lists do not.*

RELATED PRODUCTS CHECKLIST

We wondered whether there would be enough material to support a regular column listing products that support the VisiCalc® program. It seems there certainly is. In fact, we can't include in this issue all of the products we learned about while we researched the topic. So you can count on seeing more listings in the November/December issue.

The information contained in this column has been gleaned from a variety of sources including dealers, press releases, manufacturers, and advertising. Special thanks go to Carolyn Goates, formerly of CPU Corporation of Charlestown, Massachusetts, now with Apple Computer in Cupertino, California, for her help in gathering the information.

No one at Software Arts, Inc. has reviewed or examined any of these products to determine whether they perform as claimed or listed below. An inclusion in this listing does not constitute an endorsement by Software Arts, Inc. of that product. We also wish to point out that if any of the products listed alter, patch, or copy the VisiCalc program, they may be in violation of your user licensing agreement.

Since we have not reviewed any of the products, we cannot provide you with any more information than is contained in the list. If you want to learn more or you wish to purchase any of the products listed, contact your dealer first. Although we have identified distributors in some listings, many distributors do not accept orders from individuals for single copies of software or pieces of hardware.

SOFTWARE

Only software that claims to complement the VisiCalc program will be listed.

Templates

QUIKCALC

- Runs on Apple

Real estate templates to use with the VisiCalc program.

Developed and distributed by Simple Soft, Inc., 480 Eagle Drive, Suite 101, Elk Grove, IL 60007.

VISITEMPS

- Runs on Apple II

Application overlays to use with VisiCalc. Available for Income Tax Preparation, Common Business Applications, Home & Family Applications, and Sports Applications. A VisiCalc Tutor is also available.

Distributed by Exec Systems, PO Box 192, Clinton, MD 20735.

Graphics

Business Graphics ///

- Runs on Apple ///

Graphics program to load, plot and analyze VisiCalc data, as well as Basic and Pascal text files.

Distributed by Apple Computer Co., 10260 Bandley Drive, Cupertino, CA 95014.

Chart-Master

- Runs on Apple II

Business graphics plotted in eight colors, can plot selected rows and columns of VisiCalc models, create, edit and store graphs, diagrams, text pages and abstract graphics.

Distributed by Decision Resources, 44 White Birch Road, Weston, CT 06883.

Suggested retail price: $375.

Graph

- Runs on Apple II

Graphics software package, can interface with the VisiCalc program and other Apple software.

Distributed by Software Publishing Corp., 1901 Landings Drive, Mountain View, CA 94043.

Suggested retail price: $125.

Trend-Spotter

- Runs on Apple II
- Supports DIF™ format

Graphs and plots can be produced from data files, including VisiCalc files, and then manipulated.

Developed by Friend Information Systems.

Distributed by Software Resources, Inc., 44 Brattle Street, Cambridge, MA 02138.

Suggested retail price: $175.

Visichart

- Runs on Apple II

Data can be charted or read from existing curves. Performs numerical operations on an entire data set and acquires data from an analog-to-digital converter.

Produced by Interactive Microware, Inc., PO Box 771, State College, PA 16801.

Suggested retail price: $39.95.

VisiTrend/VisiPlot

- Runs on Apple II
- Supports DIF format

Plotting facility, combined with a trend-spotting facility. Uses VisiCalc files.

Developed by Lotus Development, 180 Franklin Street, Cambridge, MA 02139.

Distributed by VisiCorp, 2895 Zanker Road, San Jose, CA 95134.

Suggested retail price: $300.

SUBMITTING ARTICLES TO SATN

Articles and pointers submitted to SATN are considered for acceptance based on their quality and interest to SATN readers.

We would prefer to see queries first. If you submit an article without querying us first, include a stamped, self-addressed envelope (or mailer for diskettes). If you do not enclose an SSAE or mailer, we cannot return your material. Prospective authors may send for the *SATN Writer's Guide.*

Address all queries, articles, or requests for the guide to:

SATN
Editorial Department
P.O. Box 494
Cambridge, MA 02139

Communications

Micro-Courier

- Runs on Apple II

Charts, graphs, correspondence, VisiCalc files, or programs transmitted automatically between Apple computers over standard telephone lines.

Developed by Microcom, 89 State Street, Boston, MA 02109.

Distributed by Apple Computer, 10260 Bandley Drive, Cupertino, CA 95014.

Suggested retail price: $250.

TRANSEND

- Runs on Apple II

Data communication software, to send and receive VisiCalc reports and other programs, files, or databases. Provides Apple-to-Apple or Apple-to-mainframe communication. Three versions.

Distributed by SSM Microcomputer Products, Inc., 2190 Paragon Drive, San Jose, CA 95131.

VisiFactory

- Runs on Apple II

Data mover for VisiCalc and The Data Factory files; moves the data in either direction, manipulates it in either program, and stores it in either program.

Distributed by Micro Lab, 2310 Skokie Valley Road, Highland Park, IL 60035.

Suggested retail price: $75.

Word Processors

Apple Writer ///

- Runs on Apple ///

Word processor which can integrate VisiCalc files into reports and documents.

Distributed by Apple Products, 10260 Bandley Drive, Cupertino, CA 95014.

The Executive Secretary

- Runs on Apple II

Word processor which interfaces with The Data Factory, Information Master and VisiCalc files.

Distributed by Aurora Systems, Inc., 2040 E. Washington Ave., Madison, WI 53704.

Suggested retail price: $250.

FROM OUR READERS

While trying to work out a job costing model at our firm there are times when the VisiCalc® program gives us ERROR. It happens when the value of one calculation falls to zero and I have to divide another calculation by the zero. Is there any way around this problem?

Jim Partridge
Blackwell Plastics, Inc.
Houston, Texas

In the strictest mathematical terms division by zero is an error. In business, though, there are times when we want a division by zero to return either a zero or the results of some calculation. In those cases you can use @IF and @ISERROR to prevent the program from returning ERROR at one location and at all locations that reference the first appearance of the "erroneous" calculation.

The general form of the expression required is:

@IF(@ISERROR(value 1), value 2, value 1)

To see how that works, enter:

```
>A1:4®>B1:2
>C1:+A1/B1
>D1:@IF(@ISERROR(A1/B1),0,A1/B1)
```

The value at C1 and D1 will be 2. Now change B1 to 0. C1 will return ERROR, but D1 will have the value 0.

ScreenWriter II

- Runs on Apple II
- Supports DIF format

Text processor which can read VisiCalc files in DIF format.

Developed by Online Systems, 36575 Mudge Ranch Rd., Coarse Gold, CA 93614.

The Word Handler

- Runs on Apple

Word processor for Apples, requires no additional boards, has two character formats to choose from. Can read and use VisiCalc files.

Distributed by Silicon Valley Systems, 1625 El Camino Real, #4, Belmont, CA 94002.

SCRIPSIT II & SCRIPSIT PLUS

- Runs on Radio Shack Model II

Word processors, merge and load VisiCalc files.

Distributed by Radio Shack Division of Tandy Corporation, 1800 One Tandy Center, Fort Worth, TX 76102, and in Radio Shack stores.

Suggested retail price: $199.

Other Software

Apple VisiCalc Info Printer

- Runs on Apple II

Lister for the formats, formulas, labels and other VisiCalc grid elements. Files can be listed in either row or column sequence and may be sorted alphabetically.

Distributed by Micro-SPARC Systems, Department P, Box 325, Lincoln, MA 01773.

DBCalc

- Runs on Apple II or II Plus

Database interface which retrieves data from The Accountant's database and creates a VisiCalc file.

Distributed by Decision Support Software, 1438 Ironwood Drive, McLean, VA 22101.

Suggested retail price: $20. (with The Accountant, $109.95)

Utility Pak I

- Runs on Apple II
- Supports DIF format

DB Master format files and Apple text files can be linked with 16-sector VisiCalc files.

Developed and distributed by Stoneware Micromputer Products, Inc., 50 Belvedere St., San Rafael, CA 94901.

Suggested retail price: $99.

VC-Loader

- Runs on Apple II Plus
- Supports DIF format

Text or print files can be converted into DIF format to be exchanged with VisiCalc, VisiTrend, or VisiPlot programs.

Distributed by Micro Decision Systems, PO Box 1392, Pittsburgh, PA 15219.

Suggested retail price: $65.

HARDWARE

Only hardware that runs the VisiCalc program, or that enhances it, will be listed. We will not list 16 Kbyte RAM boards, as there are too many to cover.

Numeric Keypad

Keypad to use with Apples. Enters numbers, arithmetic operations and VisiCalc commands.

Available from The Keyboard Company, 7151 Patterson Drive, Garden Grove, CA 92641, or from dealers.

Suggested retail price is $149.95.

Videx Function Strip

Dedicated keys to define often used VisiCalc commands and filenames, so you can call them with a single keystroke. Used with Apple II.

Available from Videx, Inc., 897 NW Grant Ave., Corvallis, OR 97330.

Suggested retail price: $79.00.

TRAINING

Using VisiCalc . . . A "Hands On" Approach

Two audio cassettes and a floppy disk containing training models to help you learn to use VisiCalc easily and efficiently. Includes instructions to get an Apple II Plus up and running (but the material is not restricted to Apple II Plus users). Also includes information on various add-ons and programs which interface with VisiCalc.

Developed and distributed by McMullen and McMullen, Inc., Jefferson Valley, New York 10535.

To submit information on products (software or hardware) that support either the VisiCalc program or the DIF format for data exchange, send us the name of the product; what it does; who developed it; who markets and distributes it; the suggested retail price; if it is software, what machine(s) it runs on and whether it supports the DIF format. Send the information to:

SATN Products Column
PO Box 494
Cambridge, MA 02139

LOOKING AHEAD

The next issue of SATN begins Volume II. Some topics for future issues include calculating ACRS depreciation, using the VisiCalc® Program with selected word processing, database, and plotting programs, and more on using @AND, @OR, and @IF.

If your subscription began with Volume I, No. 1, this issue of SATN marks the time to think about renewing.

The cost of your regular SATN subscription hasn't changed. Six issues are still $30.00 in the US, or $50.00 outside the US. As a special offer, you can order 12 issues for a reduced price: $50.00 in the US, or $90.00 outside the US.

Send your subscription order and check, MasterCard, or VISA information to:

SATN Subscriptions
P.O. Box 815
Quincy, MA 02169

Don't let your advanced lessons in VisiCalc fall by the wayside. Act now!

INDEX TO VOLUME 1

Pointer p.6 #5
/E to Recalculate p.8 #4
/F * p.2 #5
/GRM p.11 #2
/M Pointer p.5 #2
/R Pointer p.4 #1
/S#S p.3 #3
/SS p.5 #2
/W and /WS p.11 #3
@AND p.2 #6
@AVERAGE p.4,5 #5; p.8 #6
@CHOOSE p.4 #1; p.5 #4
@COUNT p.8 #6
@EXP p.9 #5,
@IF p.2,4 #5; p.5 #3; p.7 #1; p.7 #4; p.3 #6
@INT p.3,7 #1; p.9 #2; p.5 #3; p.3,4 #5
@ISERROR p.10 #6
@ISNA p.11 #5
@LOOKUP p.1 #1; p.2, #1; p.5 #1
@LN p.9 #5
@MAX p.3 #1
@NPV p.9 #4
@OR p.2 #6
@SUM p.2 #2
Adding Entries to a Range p.7 #3
Annuity Due p.2 #4
Annuity, Ordinary p.2 #4
Boolean Operators p.1 #6
Compounding Single Sums p.3 #4
Constructing a Receivables Form p.7 #3
Constructing a Scheduling Form p.8 #3
Date Arithmetic p.1 #3
DIF p.4 #1
Fibonacci Numbers p.7 #1
Forward References p.3 #2 (also see the VisiCalc Planner)
General Formula for:
  Annuity Due p.2 #4
  Date Arithmetic p.4 #3
  Finding Interest Rates p.7 #5
  Level Payment to Amortize a Loan p.2 #2
  Level Payment to Amortize a Loan (Radio Shack Model I) p.5 #4
  Ordinary Annuity p.2 #4
  Rounding to Nearest Penny p.7 #1
  Sinking Fund p.5 #4
Golden Ratio p.7 #1
Group Weight p.3 #5
Interest Rates p.5 #5
Internal Rate of Return, Newton's Method p.11 #4
Internal Rate of Return, Table p.9 #4
IRAs p.1 #4
IRAs, Planning the Sheet p.5 #6
Level Payment Formula p.2 #2
Manual Recalculation p.11 #2
Modulo p.9 #2; p.9 #3; p.12 #5
Part Numbers p.9 #5
Pointers
  # Pointer p.6 #5
  /E to recalculate p.8 #4
  /M command p.5 #2
  /W and /WS p.11 #3
  @COUNT and @AVERAGE p.8 #6
  Adding entries to a range p.7 #3
  Manual recalculation p.11 #2
  Range-into-range replication p.4 #1
  Sixteen-sector VisiCalc p.8 #1
  VC displayed on start up p.8 #2
Political Forecasts p.1 #5
Planning Your VisiCalc Sheet p.5 #6
Printing VisiCalc Formulas p.5 #2
Printing VisiCalc Formulas for Apple /// Users p.8 #3
Range-into-range Replication p.4 #1
Recreational Math
  Cryptarithms p.8 #2
  Double Your Number p.12 #3
  Fibonacci Numbers p.7 #1
  Solution to "Send More Money" p.12 #3
  Sports Standings p.11 #2
Rounding to Nearest Penny p.7 #1
Saved Sheet p.5 #2
Schedule A of Form 1040 p.6 #4
Sinking Fund Formula p.5 #4
Sixteen-sector VisiCalc p.8 #1
Software Arts Spreadsheet Planner p.6 #6
Taxes p.6 #4
Trend Analysis p.4 #5
Turning Expressions into Data with DIF p.3 #3
Using the VisiCalc Planner p.5 #6
VisiCalc Calendar p.1 #3
VisiCalc Planner p.6 #6
Winner Takes All p.3 #5
Withholding Taxes p.5 #1

HAVE YOU MOVED?

If you have, let us know. To keep your issues of SATN coming to you regularly, we need to know your current address. Send a card or letter, indicating both your old address and your new one, to:

SATN Subscriptions
P.O. Box 815
Quincy, MA 02169

Remember to allow about 4 weeks for the change to take effect.

BACK ISSUES AVAILABLE

Back issues of SATN are now available. The cost is $6.00 per copy for orders mailed within the USA. If your order is to be mailed outside the USA, add $2.50 for shipping and handling to the full price of the order.

To order back issues, please send your name and address, a list of the specific issues of SATN you want (by Volume and Issue numbers), and a check or money order payable to Software Arts, Inc., to:

SATN Back Issues
P.O. Box 494
Cambridge, MA 02139